REPORT DOCUMENTATION PAGE	*Form Approved* *OMB NO. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comment regarding this burden estimates or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE July 1996	3. REPORT TYPE AND DATES COVERED	
4. TITLE AND SUBTITLE Fractal Estimation of Flank Wear in Turning Part-2: Implementation Details		**5. FUNDING NUMBERS** DAAH04-96-1-0082	
6. AUTHOR(S) S. Bakkapatnam, S.R.T. Kumara, A. Lakhtakia			
7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(ES) Center for Multivariate Analysis 417 Classroom Building The Pennsylvanica State University University Park, PA 16802		**8. PERFORMING ORGANIZATION REPORT NUMBER**	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Research Office P.O. Box 12211 Research Triangle Park, NC 27709-2211		**10. SPONSORING / MONITORING AGENCY REPORT NUMBER** ARO 30529.92-MA-SDI	

11. SUPPLEMENTARY NOTES

The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release: distribution unlimited.	**12 b. DISTRIBUTION CODE**

13. ABSTRACT *(Maximum 200 words)*

Abstract

In Part-1 of this paper we described the theoretical foundations of fractal estimator development. The implementation aspects of the fractal estimator for on-line tool wear estimation in the turning process is described in this part. The results show that this paradigm can be successfully used for real-time quality control in manufacturing.

19961023 242

14. SUBJECT TERMS	**15. NUMBER IF PAGES** 25
	16. PRICE CODE

17. SECURITY CLASSIFICATION OR REPORT UNCLASSIFIED	**18. SECURITY CLASSIFICATION OF THIS PAGE** UNCLASSIFIED	**19. SECURITY CLASSIFICATION OF ABSTRACT** UNCLASSIFIED	**20. LIMITATION OF ABSTRACT** UL

FRACTAL ESTIMATION OF FLANK WEAR IN TURNING
PART-2: IMPLEMENTATION DETAILS

Satish Bukkapatnam
Graduate Student in
Industrial and Manufacturing Engineering

Soundar R. T. Kumara*
Professor of Industrial
and Manufacturing Engineering

Akhlesh Lakhtakia
Associate Professor of Engineering Science and Mechanics
The Pennsylvania State University
University Park, PA 16802

Abstract

In Part-1 of this paper we described the theoretical foundations of fractal estimator development. The implementation aspects of the fractal estimator for on-line tool wear estimation in the turning process is described in this part. The results show that this paradigm can be successfully used for real-time quality control in manufacturing.

1

Contents

1 INTRODUCTION

On-line tool wear estimation is deemed essential to realize the goal of real-time quality control in the turning process. Towards realizing an on-line tool wear estimation scheme, we have concentrated our research efforts on flank wear estimation only; but we foresee a combined flank and crater wear estimation in the near future.

Conventionally, flank wear is quantified by the height of the flank wear land h_w because h_w is considered to have the maximum bearing on the flank wear (Kamarthi, 1994), and $h_w > 0.018$ inch is considered to be a good indicator that the tool is fully worn (ISO, 1972). In many industrial scenarios it takes between 5 to 20 min for a cutting tool to wear out. A viable continuous flank wear estimator should therefore have accuracies to match the values between $0.0 - 0.018$ inch over the above specified time span. Furthermore, accuracies of $\leq 10\%$ of the total range are desirable because the continuous flank wear estimates may be directly used to (i) enforce geometric adaptive control (Kamarthi, 1994), (ii) plan tool change strategies, and (iii) control tool wear rate to maintain the desired level of surface integrity, and other quality variables.

Many researchers have attempted the problem of tool wear estimation with varying degrees of success. Most of the earlier models were either (i) analytical models-based observers, (ii) empirical estimators or (iii) neural network-based estimators. However, most of these techniques explicitly or implicitly assumed the sensor signals to be predominantly harmonic with additive stochastic contaminants. Furthermore, most of the observers used sensor signals sampled at low-frequencies to develop Kalman filters for tool wear estimation, thereby ignoring the overall variations in the dynamics of the turning process (referred to as turning dynamics).

Recently, a novel neural network-based estimator was developed in (Kamarthi, 1994). This method overcomes the need to assume an explicit mathematical model for tool wear, and at the same time captures variations in turning dynamics. But the architecture of the resulting neural network was fairly complicated. Moreover, the procedures of training, testing and installation thereof were computationally intensive. The methodology we have proposed in Part-1, called fractal estimation, is aimed at simplifying the neural network architecture while at the same time overcoming the need to assume an explicit tool wear model.

The fractal estimation methodology we put forth to develop a *fractal estimator* involved extracting the fractal properties of sensor signals to compute flank wear estimates (as described in Part-1). Our earlier experimental characterization of the turning dynamics as being low-dimensional chaos (Bukkapatnam et al., 1995) formed the basis for fractal estimation methodology.

The overall methodology involves (i) development phase, where a neural network is trained with the signal features, and (ii) operation phase where the trained neural network—the fractal estimator—is used for on-line estimation. The development phase, shown in Figure 1(a), involves the following four steps:

1. **Experimentation:** The purpose of the experiments is to obtain the actual tool wear values and the on-line sensor data, so that the features extracted from the latter can be used to train a supervised neural network for wear estimation, with the actual wear as the target.

2. **Signal separation:** It is a preprocessing operation in which the signal contaminants are removed so that the values of the extracted signal features become more accurate.

3. **Feature extraction:** It involves extracting the combined fractal dimensions of different sensor signals, such that the values of the extracted fractal dimensions contain all the necessary information to estimate flank wear.

4. **Recurrent neural network design:** It involves (i) using the extracted features to develop exemplar patterns (ii) developing a neural network having a suitable architecture, and (iii) training the neural network through backpropagation algorithm to learn the relationships associating the features with tool wear.

The operation phase, shown in Figure 1(b) consists of (i) on-line acquisition of the sensor data during the turning process, (ii) signal separation, (iii) feature extraction, and (iii) input pattern preparation and presentation thereof to the trained neural network to obtain flank wear estimates.
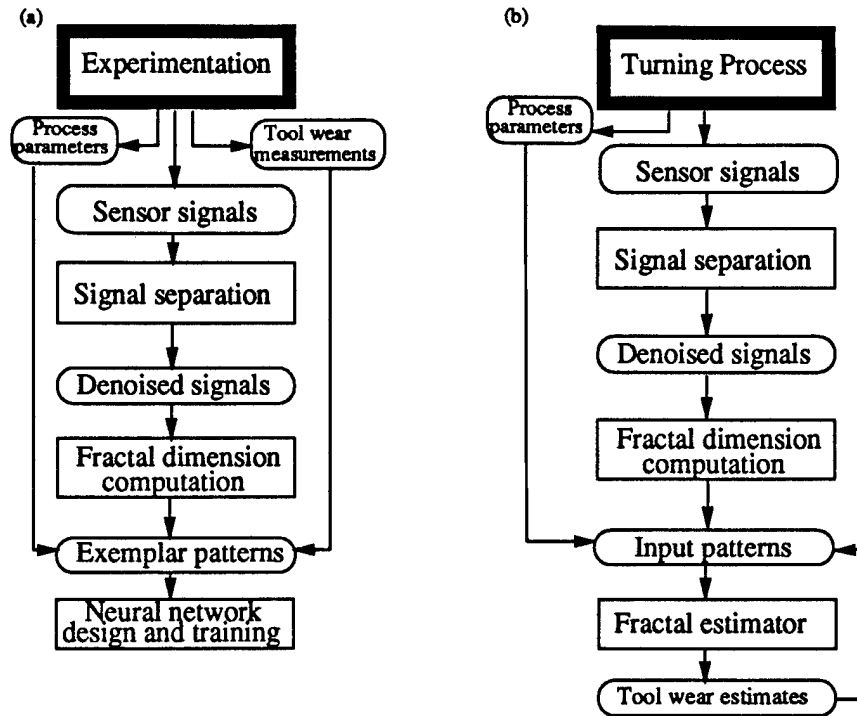
Figure 1: Installation procedure for fractal estimator: (a) Development phase (b) Operation phase.

In this part of the paper, the implementation details and results of performance evaluation of fractal estimator are presented. This paper is organized as follows: Step-wise implementation of the development phase of the fractal estimator is presented in Section 2; in Section 3 we present the results of rigorous performance evaluation; and in Section 4 we discuss the issues involved in the operation phase of fractal estimator. Finally we present our conclusion and the future work. The results of the performance testing, we hope, will spur industrial application of our paradigm.

2   IMPLEMENTATION AND RESULTS

In this section we present a thorough step-wise implementation details of the development phase of fractal estimator. This section is accordingly divided in to four subsections corresponding to the four steps of the development phase: (a) experimentation, (b) signal separation, (c) feature extraction and (d) neural network design.

2.1   Experimentation

The basic aim behind performing machining experiments is to obtain the actual tool wear values and the on-line sensor data, so that the features extracted from the latter may be used to train a supervised neural network for flank wear estimation.

The experiments were conducted on a 20 HP LeBlond heavy duty lathe. The workpieces were made of 36 *inch* × *ϕ*7 *inch* SAE 6150 Cr-V steel, and the tool inserts were uncoated carbide grade K68 with geometric specification SPG-422. Three different on-line sensors were used—(i) a 3-axis Kistler Z3392/b piezo-electric dynamometer for measuring cutting, feed and thrust forces, (ii) two PCB accelerometers to measure vibration signals along main and feed directions, and (iii) a SE-900 MWB wide bandwidth AE sensor to measure RMS AE signals. The force signals were sampled at 3 kHz frequency, vibration signals at 26 kHz, and acoustic emission signals at 1 MHz.

A 5 × 5 full factorial experimental design consisting of five cutting speeds (100, 130, 160, 190 and
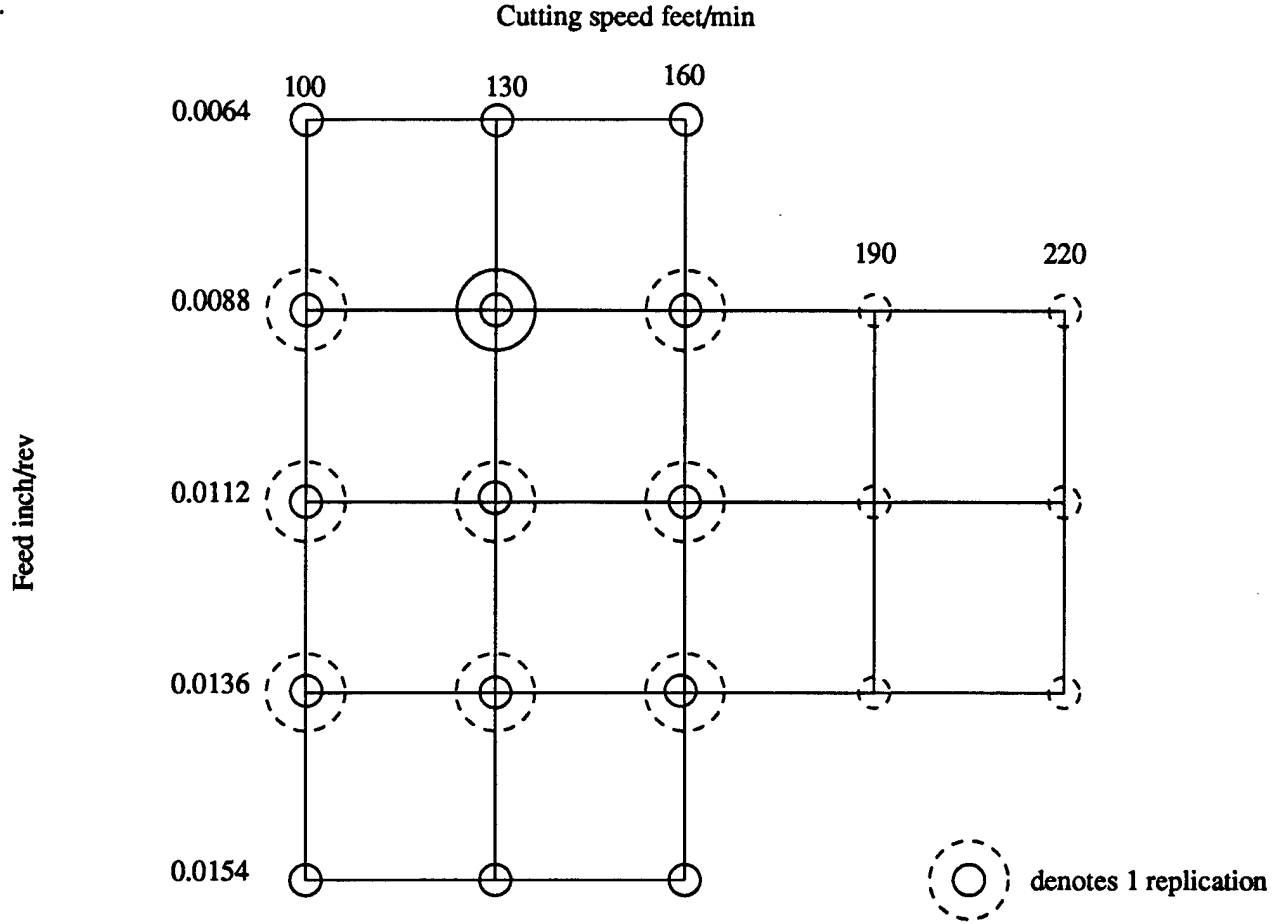
4

Cutting speed feet/min

Figure 2: Design of experiments: Solid and dashed outlines of circle indicate whether that the exemplar patterns corresponding to that design point are used for training or testing .

220 feet/min), and five feeds (0.0064, 0.0088, 0.0112, 0.0136 and 0.0154 inch/rev) was used. The depth of cut was kept constant at 0.05 inch. The experimental design is shown in Figure 2. At every design point, a fresh cutting edge was used to perform the turning operation and time-series data (TSD) of length 4096 was collected from sensor signals at regular intervals of 1 minute till the tool wore down (i.e., $h_w > 0.018$ inch). This design ensures that the process parameter space is sampled reasonably uniformly in the operating range The uniform sampling in turn improves the chances of exemplar patterns extracted from the measured TSD to be *rich* in the input space.

Figure 3 shows a representative measured TSD corresponding to the force and vibration sensor signals collected from an experiment conducted at cutting speed = 130 feet/min, feed = 0.0152 inch/rev and tool wear h_w = 0.0074 inch. We observed that the forces and vibration sensor signals exhibit low-dimensional chaos and were fairly stationary (Bukkapatnam et al., 1995). Even though we collected acoustic emission during experiments, it was not considered because the TSD was highly transient and corresponded to a complex higher order dynamics. All the measured TSD from force and vibration sensors were then subjected to signal separation.

2.2 Signal separation

Signal separation of the measured TSD was performed using wavelet transforms. Our signal separation scheme, shown in Figure 4, consists of four steps:

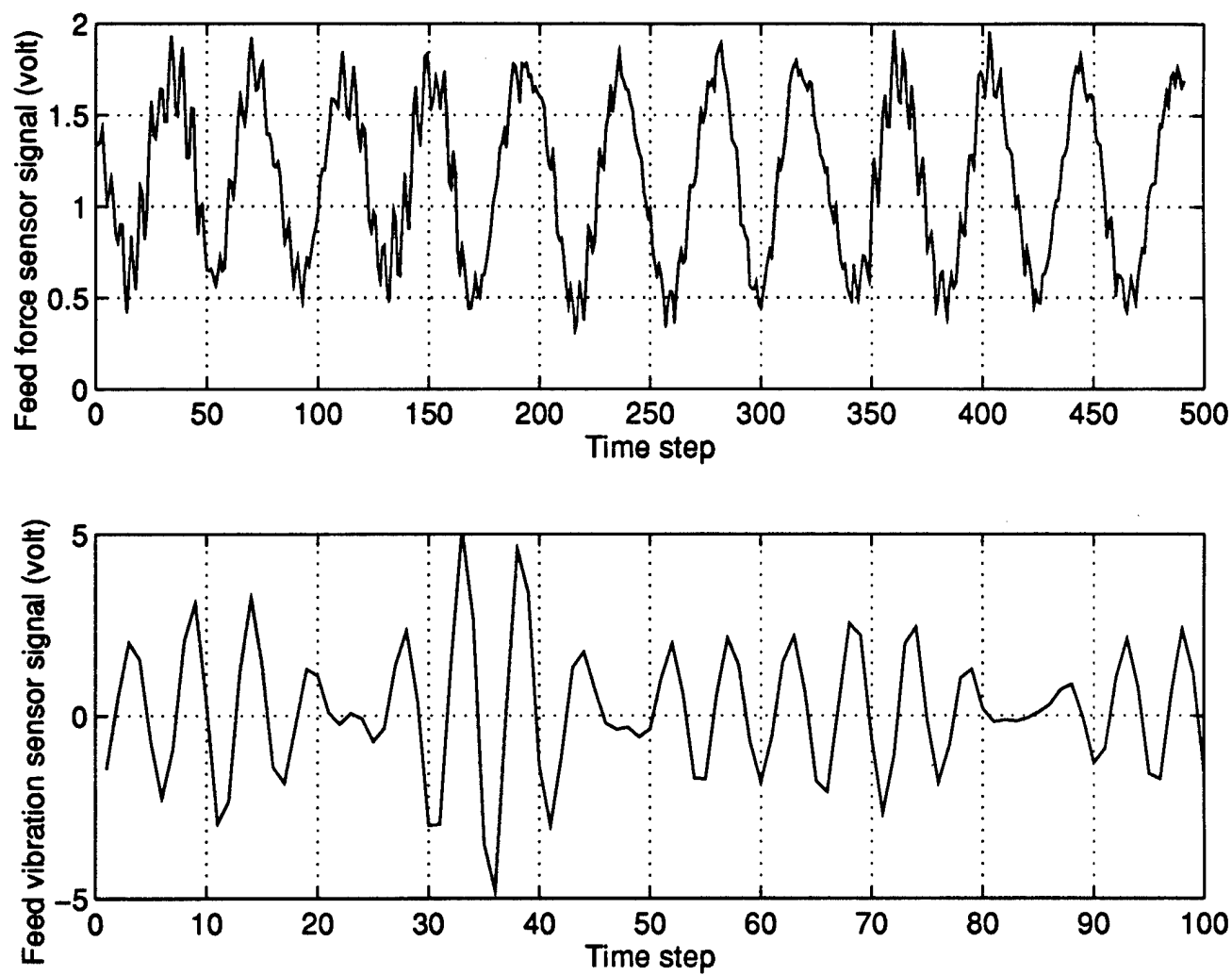   (i) representing a measured TSD in terms of wavelets and extracting the wavelet coefficients,

Figure 3: Representative TSD of force and vibration sensors from an experiment conducted at cutting speed = 130 feet/min, feed = 0.0152 inch/rev and h_w = 0.0074 inch.
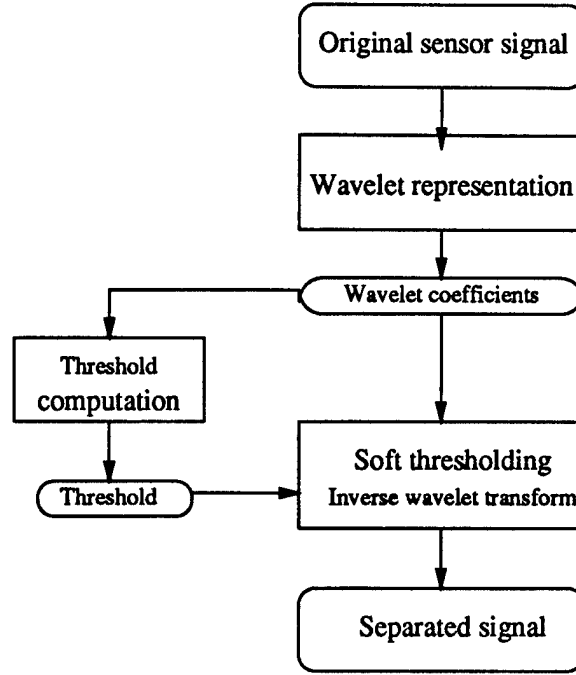
Figure 4: Block diagram of wavelet transform-based signal separation.

(ii) computing the threshold from these coefficients,

(iii) performing soft thresholding of the coefficients, and

(iv) inverse transforming these thresholded coefficients to get the separated signal.

From an implementation standpoint, the wavelet transform-based signal separation is equivalent to a filtering operation in which the measured sensor signals are passed through a filter-bank of high- and low-pass filters with different center frequencies and bandwidths determined by a pyramidal algorithm. The block diagram of the pyramidal algorithm implementation is shown in Figure 5.

The input signal is first transformed (projected) into the two orthogonal subspaces spanned by $a_{1,k}$'s and $b_{1,k}$'s corresponding to low-pass filter LP and high-pass filter HP respectively. As described in Part-1, a's are referred to as scaling coefficients and b's as wavelet coefficients.

The wavelet coefficients are subjected to thresholding after which they are *upsampled* by 2 and then inverse transformed by the filter IHP. The scaling coefficients are first *downsampled* by 2 and then filtered through LP and HP once again to obtain coefficients corresponding to the next lower resolution. The scaling coefficients are inverse transformed using the filter ILP and added to the filterates from IHP of the same level to obtain separated signal of resolution upto that level. The architecture of HP-IHP and LP-ILP pairs are analogous to analysis-synthesis parts of a quadrature mirror filter (Goldberg, 1993).

The implementation details of signal separation are as follows: Every measured TSD—a representative of which collected from main vibration sensor at cutting speed = 160 feet/min, feed = 0.0136 inch/rev, and tool wear = 0.0024 inch is shown in Figure 6(a)—was subjected to a discrete-time wavelet transform using Debauchies D_4 wavelets. The resulting wavelet coefficients are shown in in terms of a multiresolution representation (Mallat, 1989) in Figure 7. This figure shows the dominant coefficients in each scale. For instance, in the row corresponding to 2^3-frequency scale, there are 4 equally spaced positive wavelet coefficients and 4 negative coefficients.

Next the standard error for each scale j of the wavelet coefficients was computed, thereby arriving at the threshold τ_j^j at each level as given by (17) of Part-1 of this paper. The threshold values for the representative TSD are shown in Table 1. The values of τ_j^j hover between 0.3936 and 0.9391 corresponding to an approximately 6% reduction in the signal energy. The individual wavelet coefficients
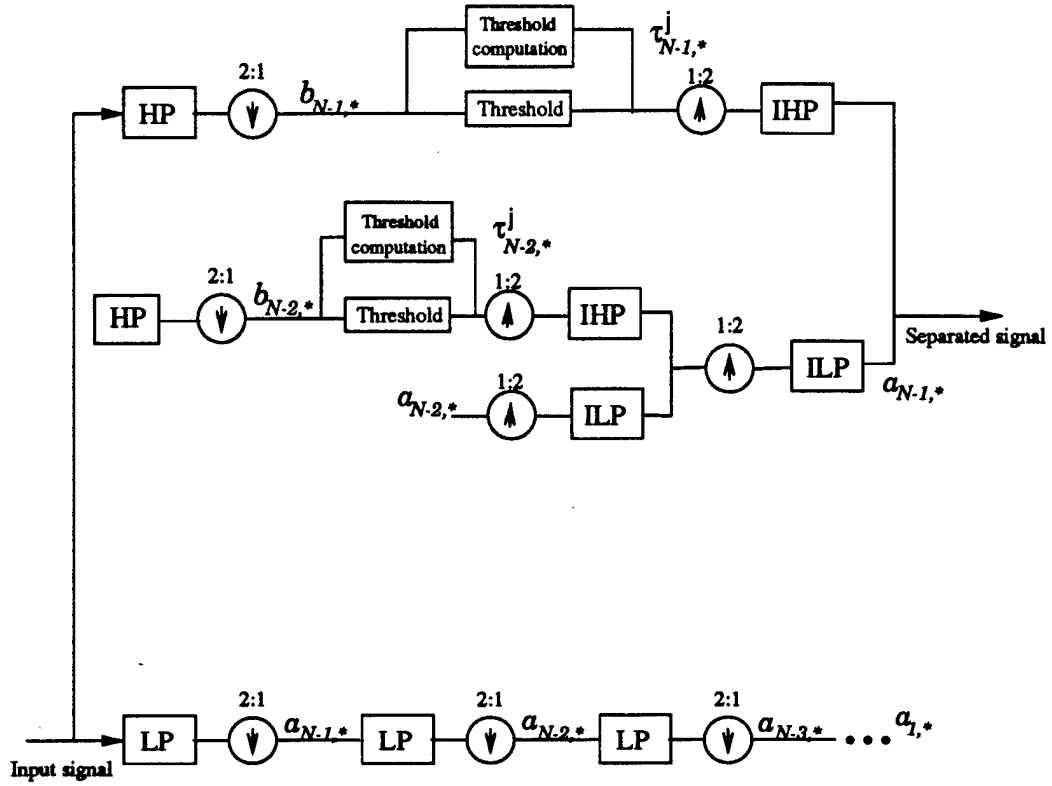
7

Figure 5: Filter-bank implementation of signal separation ('*' refers to all possible values of the second subscript **k**).

Table 1: Results of soft threshold computation for the representative TSD

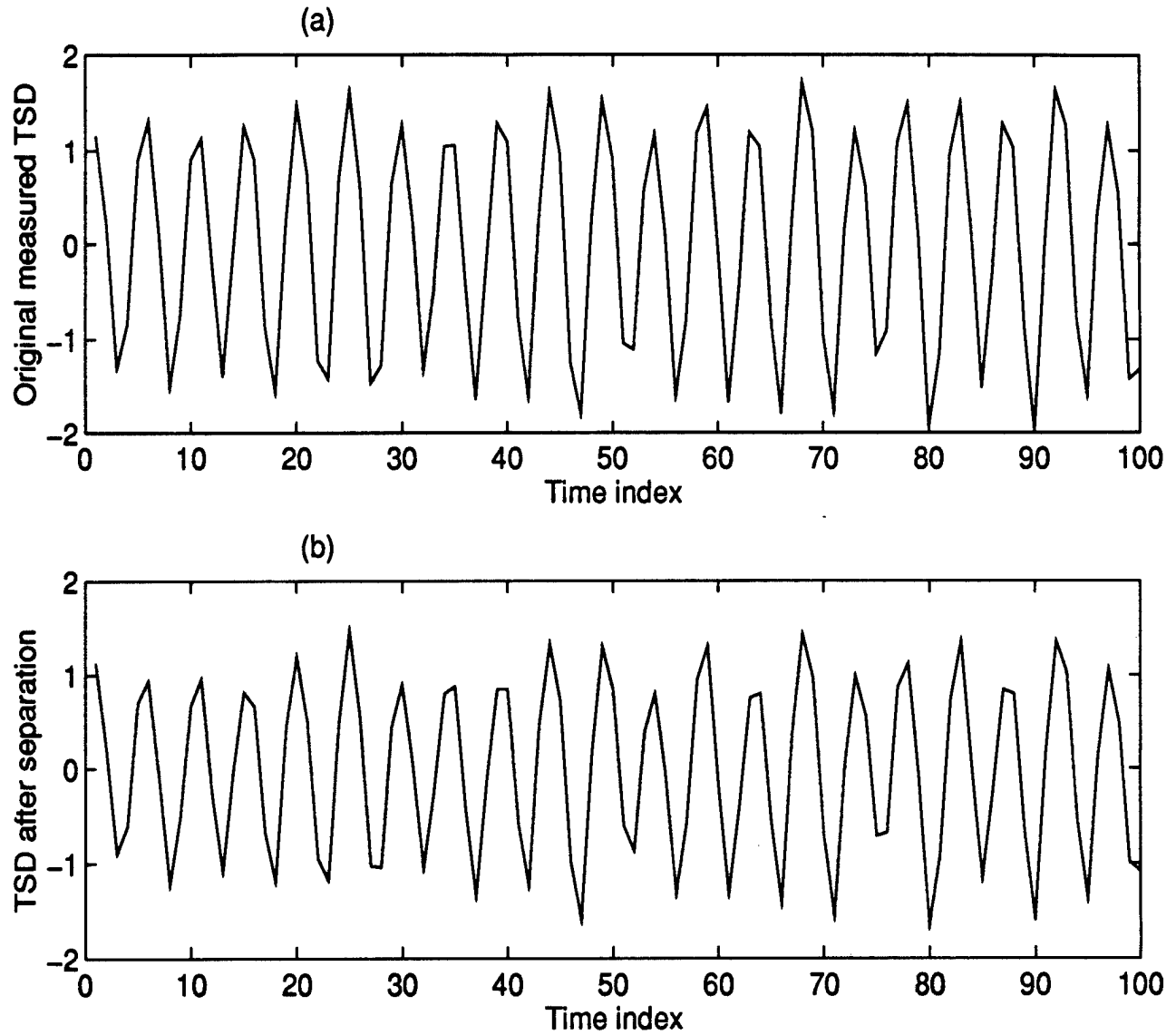| Scale	Standard error	τ_j^j
11	0.10605	0.6363
10	0.12872	0.7723
9	0.09575	0.5745
8	0.11908	0.7145
7	0.12892	0.7350
6	0.11245	0.6747
5	0.15652	0.9391
4	0.06560	0.3936
3	0.10255	0.6153
2	0.11418	0.6851
1	0.13510	0.8106

Figure 6: Comparison of signal before and after separation: (a) Original measured TSD (b) TSD after separation.
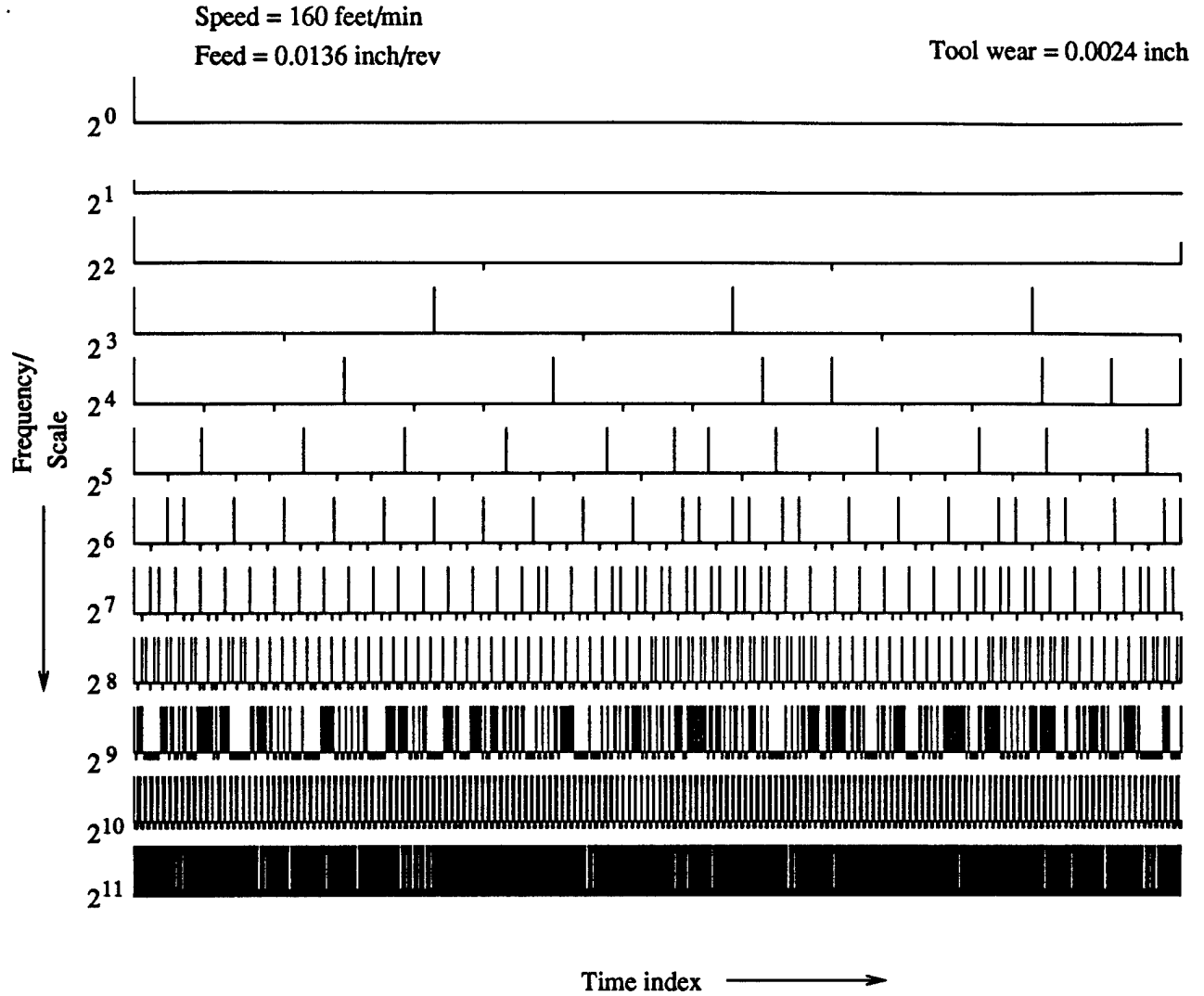
Figure 7: Multiresolution representation of the wavelet coefficients after signal separation: Vertical lines above the horizontal reference line signify positive coefficient and the ones below signify negative coefficient.

$b_{j,k}$ were then soft thresholded as given by (18) in Part-1 of this paper. The resulting values were subjected to inverse wavelet transform. The resulting TSD after separation is shown in Figure 6(b). The figure reveals that signal separation smoothens the measured TSD without depriving the latter of its essential trends. The effects of this smoothing become more clear when we look at the fractal dimension estimates shown in Figure 10. The separated signals were then subjected to feature extraction.

2.3 Feature extraction

Feature extraction essentially involved computing fractal dimensions of the measured TSD. The implementation details of feature extraction are described in this subsection.

The scalar TSD corresponding to main force, feed force, main vibration and feed vibration, obtained from each individual experimental run, were combined, after signal separation, to form 4-dimensional vector TSD. The rationale for choosing 4-dimensional vector TSD is explained in the following.

The Poincaré section plots of force and vibration sensor signals revealed an almost circular pattern (Bukkapatnam et al., 1995). We then examined the double Poincaré section plots which also
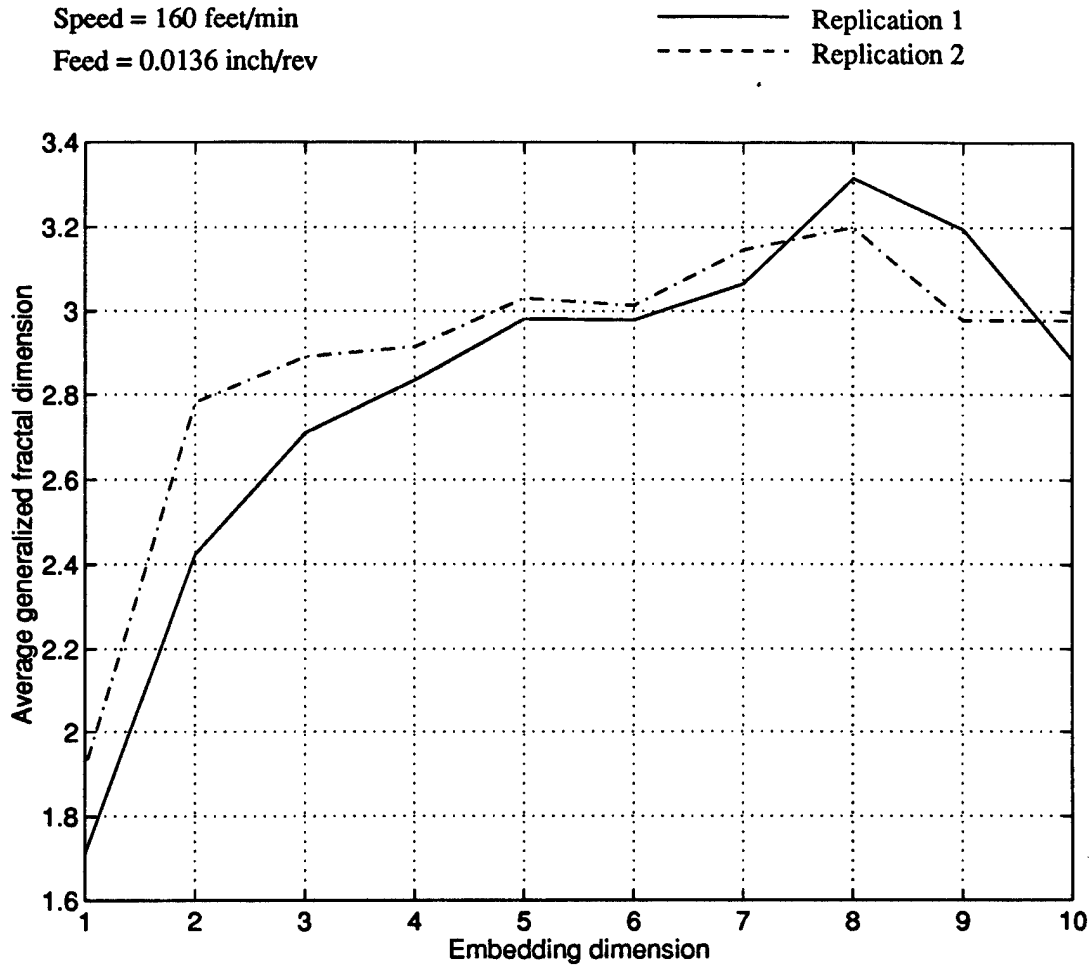
10

Figure 8: Representative plots showing the variation of average fractal dimension with embedding dimension for two experimental runs conducted using a fresh tool: cutting speed = 160 feet/min, feed = 0.0136 inch/rev.

revealed an almost circular disposition of points. These results indicate that the attractor of turning dynamics resembles a 3-dimensional torus, an object that can be embedded in a 4-dimensional state space. In other words, an embedding dimension of $d_E = 4$ is optimal for reconstructing turning dynamics.

This fact has also been verified from the variation of average[1] generalized fractal dimension with embedding dimension as shown in Figure 8. First the average generalized fractal dimension of the scalar TSD corresponding to the main force sensor data was determined to be 0.91. Next, the embedding dimension was incremented by unity by including the feed force sensor data. The average fractal dimension then was found to be 1.52. Increasing the embedding dimension by including the main vibration sensor data resulted in a fractal dimension of 2.31. However, increasing the embedding dimension beyond this point by sequentially including the feed vibration and thrust force sensor data, and lag coordinates of cutting force, feed force and vibration sensor data, did not increase the fractal dimension of the attractor of the dynamic system represented by the sensor signals. We note that the sensor signals were chosen based on an order of their physical importance determined by us.

Figure 8 shows that the value of the fractal dimension of the attractor of the dynamical system represented by the sensor signals begins to stabilize when the embedding dimension exceeds 3. Thus the

---

[1]Average refers to the statistical mean of fractal dimensions obtained over the experimented combinations of cutting speeds and feeds.
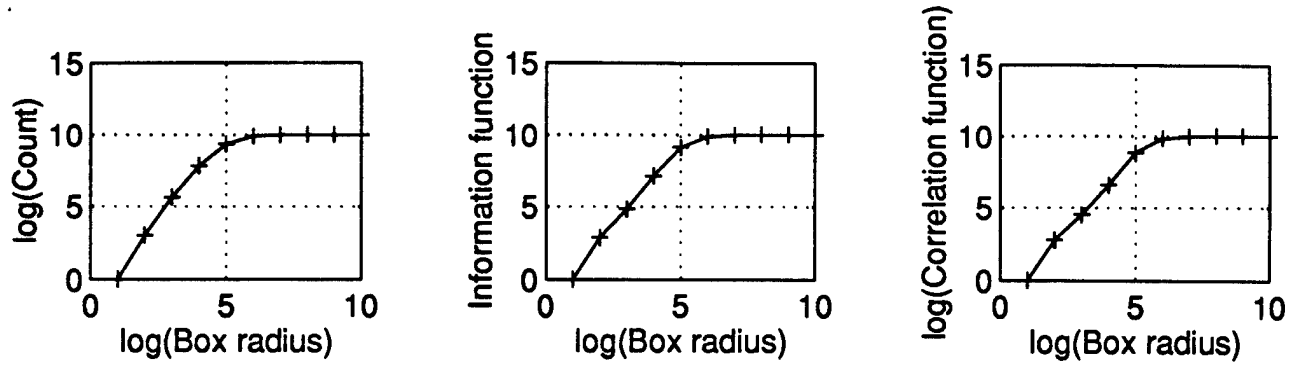
Figure 9: Representative log-log plots to obtain fractal dimensions.

average fractal dimension of the attractor was fixed at 2.63 and the optimal embedding dimension at 4. Based on this observed trend of stabilization of fractal dimension values, we may conclude that the combination of main force, feed force, main vibration and feed vibration sensor signals is adequate to compute the fractal dimensions. Thus, the features for exemplar patterns to train the proposed neural network were extracted from a 4-dimensional TSD composed of signals from

1. main force,
2. feed force,
3. main vibration and
4. feed vibration sensors.

A modified box-counting technique proposed by Liebovitch and Toth (Liebovitch and Toth, 1989) was used to compute the capacity, information and correlation dimensions of every constructed 4-dimensional TSD. The methodology of this computation is described in Part-1. The data points of every measured TSD were scaled and shifted to lie between 0 and $m = 2^{32}$. Next, (i) number of balls $N(\epsilon)$ required to cover the *phase portrait* of the 4-dimensional TSD, and (ii) frequency of points in each of these balls $p_i(\epsilon)$ were computed. Representative log-log plots of $N(\epsilon)$ versus ϵ, $\sum_i p_i(\epsilon)^{p_i(\epsilon)}$ versus ϵ, and $\sum_i p_i(\epsilon)^2$ versus ϵ are shown in Figure 9. The plots correspond to the data obtained from the experiments performed with a fresh tool under the randomly chosen parameter settings of cutting speed = 160 feet/min, feed = 0.0136 inch/rev. The slopes of the linear portions of these graphs yield capacity, information and correlation dimension, respectively.

The plots show linear trend for more than 3 decades of ϵ. This implies that the slopes of the plots and hence fractal dimensions can be computed accurately.[2] Furthermore, the values of the fractal dimensions, calculated from the slopes of these graphs, for the representative vector TSD are $D_0 = 2.77$, $D_1 = 2.69$, and $D_2 = 2.53$.

The effect of signal separation may be understood by comparing of the log-log plots of correlation dimension corresponding to the measured sensor signal before and after signal separation. The comparison plots of Figure 10 clearly show that graph drawn for the separated signal has a sharper transition from linear to flat portions of the graph compared with the non-separated signal. This sharper transition enables a more accurate computation of fractal dimensions. Fractal dimensions thus computed were used as the signal features for tool wear estimation.

2.4 Recurrent neural network design

The purpose of the neural network is to relate the fractal dimensions of the sensor signals to tool wear. However, the results of fractal dimension computation showed a complicated, and seemingly patternless, relationship connecting the extracted fractal dimensions and tool wear. A representative plot showing the variation of the capacity dimension with wear is shown in Figure 11.

---

[2] We may note that ≥ 3 points are required to assess the linear trend in the data.
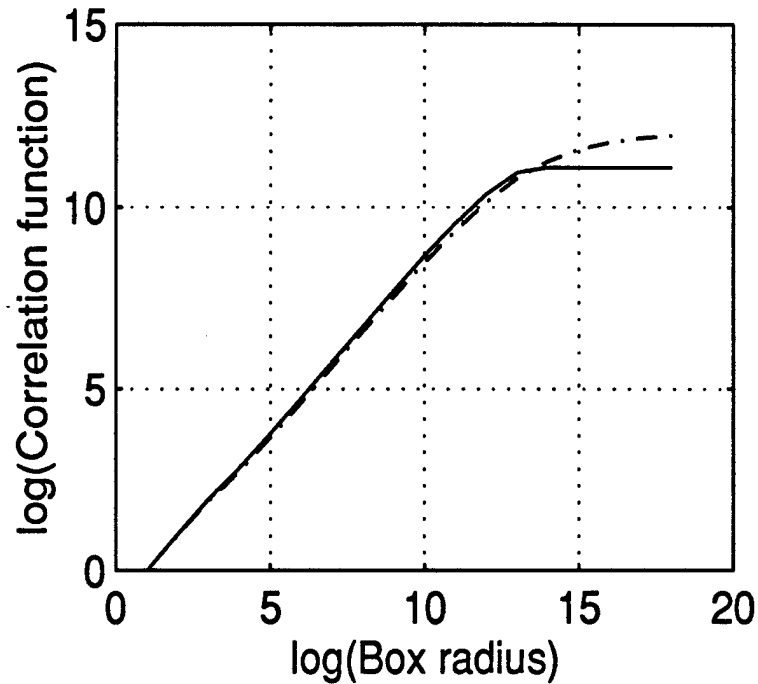
Figure 10: Effect of signal separation on the fractal dimension estimates: —- corresponds to the separated signal, and ------- corresponds to the non-separated signal. As a result of signal separation, the change of slope of the graph becomes more pronounced. This reduces the uncertainty in deciding the linear portion of the graph thus rendering the computation of the slope of the linear portion of the graph, and hence the fractal dimension estimates more accurate.
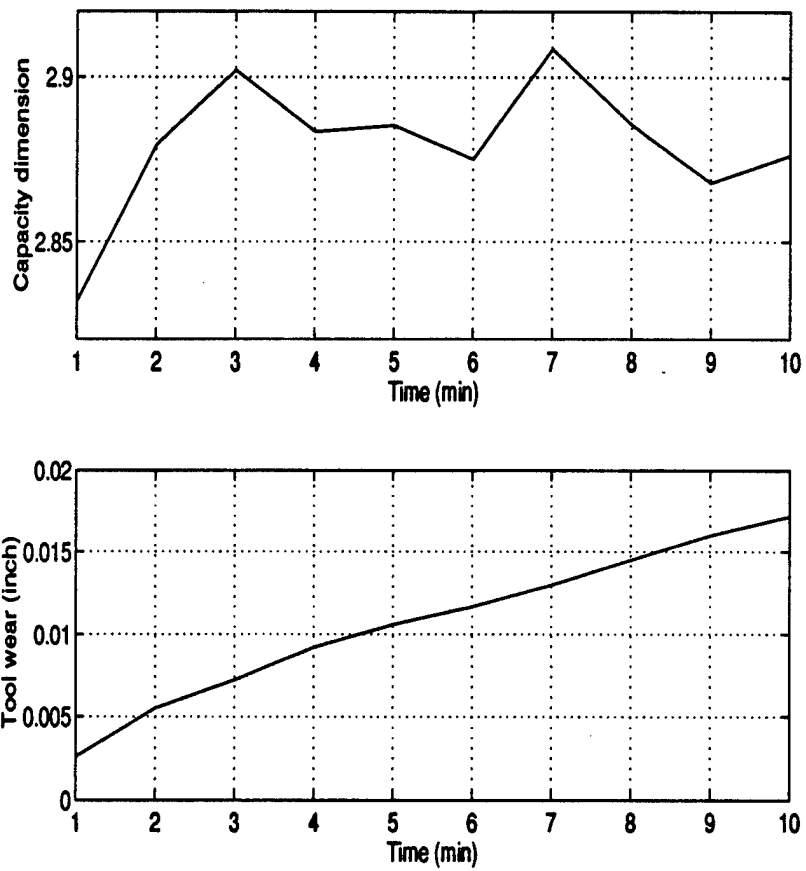
Speed = 160 feet/min
Feed = 0.0136 inch/rev

Figure 11: Variation of capacity dimension of sensor signals with tool wear.

Since the extracted fractal dimensions correspond to the system state during a specific sampling interval, information on the system history is not contained in the fractal dimension values. But, tool wear is a continuous, monotonically nondecreasing phenomenon, and thus accurate abstraction of tool wear growth requires the entire system history.

Mathematically, this relationship may be captured by a set of differential equations in an observer form as

$$\dot{\mathbf{w}} = \mathcal{G}(\mathbf{w}, t, \varpi) \tag{1}$$

$$D_q = \mathcal{H}(\mathbf{w}, t, t_w, \vartheta), \tag{2}$$

where $\mathbf{w}$ is a vector consisting of instantaneous values of h_w and its higher order derivatives, t_w is the time-window used to compute generalized fractal dimensions from sensor sensor signals, ϖ and ϑ are the dynamic and measurement contamination, and $\mathcal{G}$ and $\mathcal{H}$ are vector fields. Estimating tool wear by solving this set of nonlinear stochastic differential equations on-line (in real-time) is not practical. Therefore the use of neural networks for estimation is almost certainly a better choice.

The structure of this set of equations suggests that the tool wear estimation uses the system history. Since we are using digitized sensor signals, at the very least, the immediately previous state should be included for estimating tool wear. Hence the neural network architecture must possess some form of internal memory. This requirement may be met with by using a recurrent neural network (RNN).

An RNN based estimator should satisfy the following requirements:

1. **Identifiability:** It is defined as (i) the sufficiency of the exemplar patterns to adequately represent the input space and (ii) insensitivity of the neural network weights to parameters of the training algorithm and initial weights (Zbikowski, 1994).

2. **On-line implementability:** The preprocessing algorithms and the neural network estimates should be computed on-line. The time interval t_1 between the successive estimates should be comparable to the sampling time t_2 of the measured TSD.

3. **Stability:** Since a feedback loop connects the output to the input, an RNN is morphologically similar to an extended Kalman filter. Therefore RNN entails potential stability problems. A finite error (perturbation) either in (i) turning dynamics due to sudden tool breakage, changes in the material hardness, etc., or (ii) the measurement system due to, for example, the failure of sensors, or (iii) the computational system due to finite precision errors and other approximations, and process/model uncertainties, may result in the estimation errors growing with time. Stability in this case refers to a smooth attenuation (die-out) of the influence of small finite perturbation over time.

   However, there does not exist an analytical method to validate the stability of a generic RNN. Therefore, stability of RNN should be studied using simulated/physical performance evaluation tests.

4. **Dimensionality:** The architecture of the neural network should be parsimonious with respect to the number of nodes and weights. Furthermore the dimension of the exemplar pattern space should be consistent with the underlying input-output relationship.

5. **Robustness:** Robustness refers to the insensitivity of neural network to model uncertainties, sensor failures and finite perturbations. The difference between stability and robustness is that robustness implies little sensitivity to perturbations, model uncertainties, etc.; whereas stability refers to the attenuation of estimation errors resulting from perturbations.

Consistent with the above requirements, the following data structure of the exemplar pattern was found sufficient for learning the tool wear – sensor signal relationships:

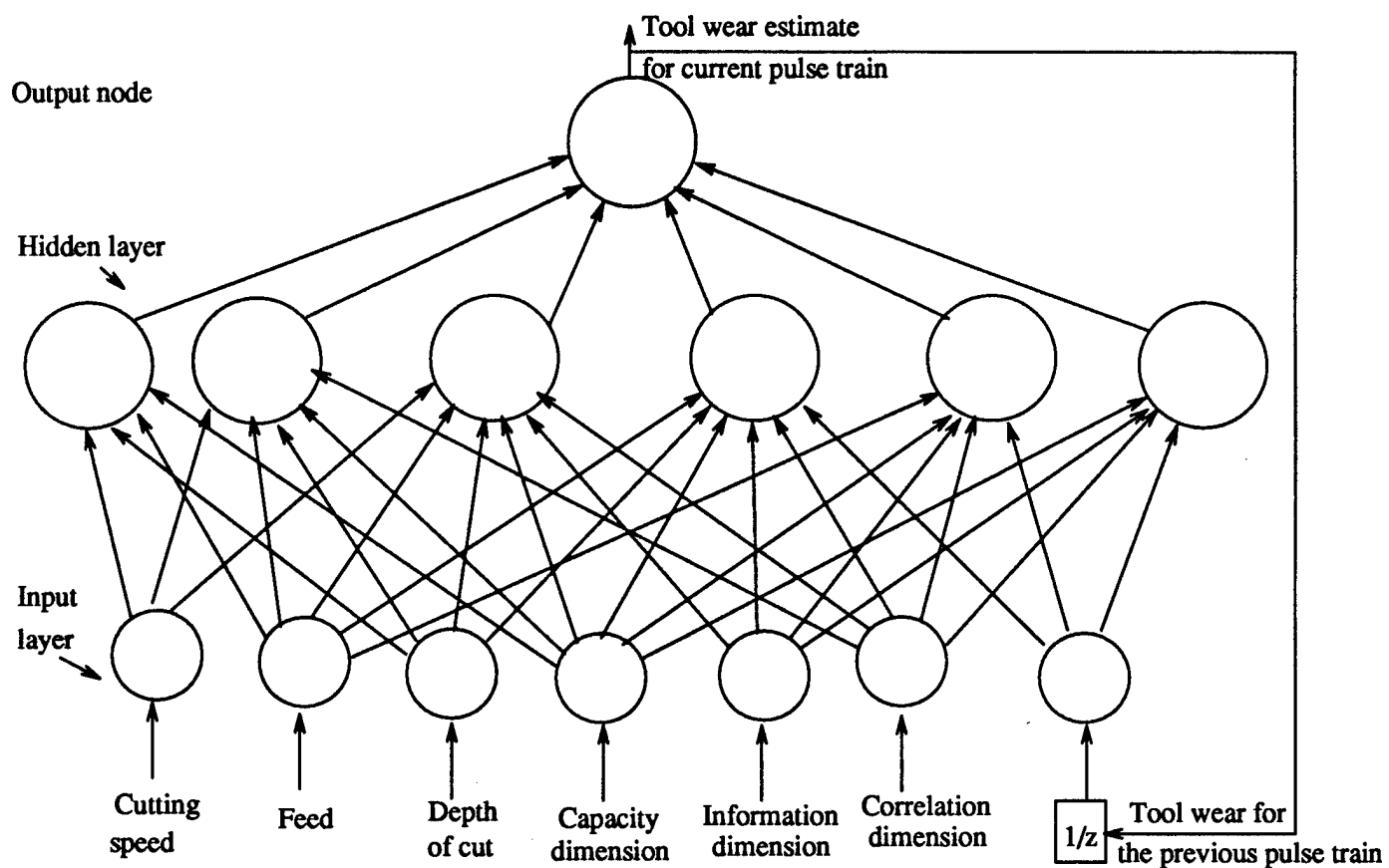1. Cutting speed
2. Feed
3. Depth of cut
4. Capacity dimension

Figure 12: Architecture of the recurrent neural network.

100.000000	0.006400	0.050000	2.203280	2.174500	2.238360	<0.000000>	0.002400
100.000000	0.006400	0.050000	3.130530	2.079740	2.795680	<0.008600>	0.010500
100.000000	0.006400	0.050000	2.736970	3.183520	3.425710	<0.010500>	0.012000
100.000000	0.006400	0.050000	2.181550	2.164290	2.219770	<0.014600>	0.015900

Table 3: Neural network training details

No. of patterns:	156
No. of input units:	7
No. of output units:	1
No. of hidden units:	6
Learning function:	Backpropagation with Momentum
No. of iterations:	10000
Learning parameter:	0.5 for first 2000 iterations, and 0.3 for the rest of training
Momentum coefficient:	0.85
Mean absolute training error:	0.0012 inch

5. Information dimension

6. Correlation dimension

7. Feedback output of the neural network, which corresponds to the tool wear estimate at the end of the previous time interval

Even though depth of cut was kept constant throughout our experiments, for the sake of completeness, we have included depth of cut also to be part of the exemplar pattern structure. The architecture of the multilayer neural network, shown in Figure 12, consisted of seven input nodes conforming to the exemplar pattern structure, and single output node corresponding to the tool wear estimate. A single hidden layer consisting of six hidden nodes was used. Sigmoidal function was used as the activation function. Sample input patterns are shown in Table 2. The details of the neural network training are summarized in Table 3.

The recurrent neural network showed a fair degree of learning and generalization over the input space consisting of the sensor signals. The convergence of the training error—defined as the root mean square of the difference between the target and the neural network output, taken over all the training patterns— is shown in Figure 13. From the figure it is evident that after training for 10000 iterations, the training error falls below 0.001 inch, implying a fairly accurate levels of estimation. Furthermore, the pattern of convergence is exponential implying that the parameters for the neural network training are adequate to capture the nonlinear relationship.

The performance of the fractal estimator developed thus was evaluated using a rigorous testing procedure. The details of the fractal estimator testing are presented in the next section.

3 PERFORMANCE EVALUATION

The trained network was tested using 150 testing patterns. A representative plot of actual versus estimated tool wear is shown in Figure 14. The experiment was conducted at 160 feet/min cutting speed and 0.0136 inch/rev feed.

The normality assumption for estimation errors were found to be reasonable as seen from the normal probability plot of Figure 15. The mean and the standard error of the estimation errors computed for the 150 testing patterns were 0.000116 inch and 0.0011 inch respectively. The 95% confidence interval
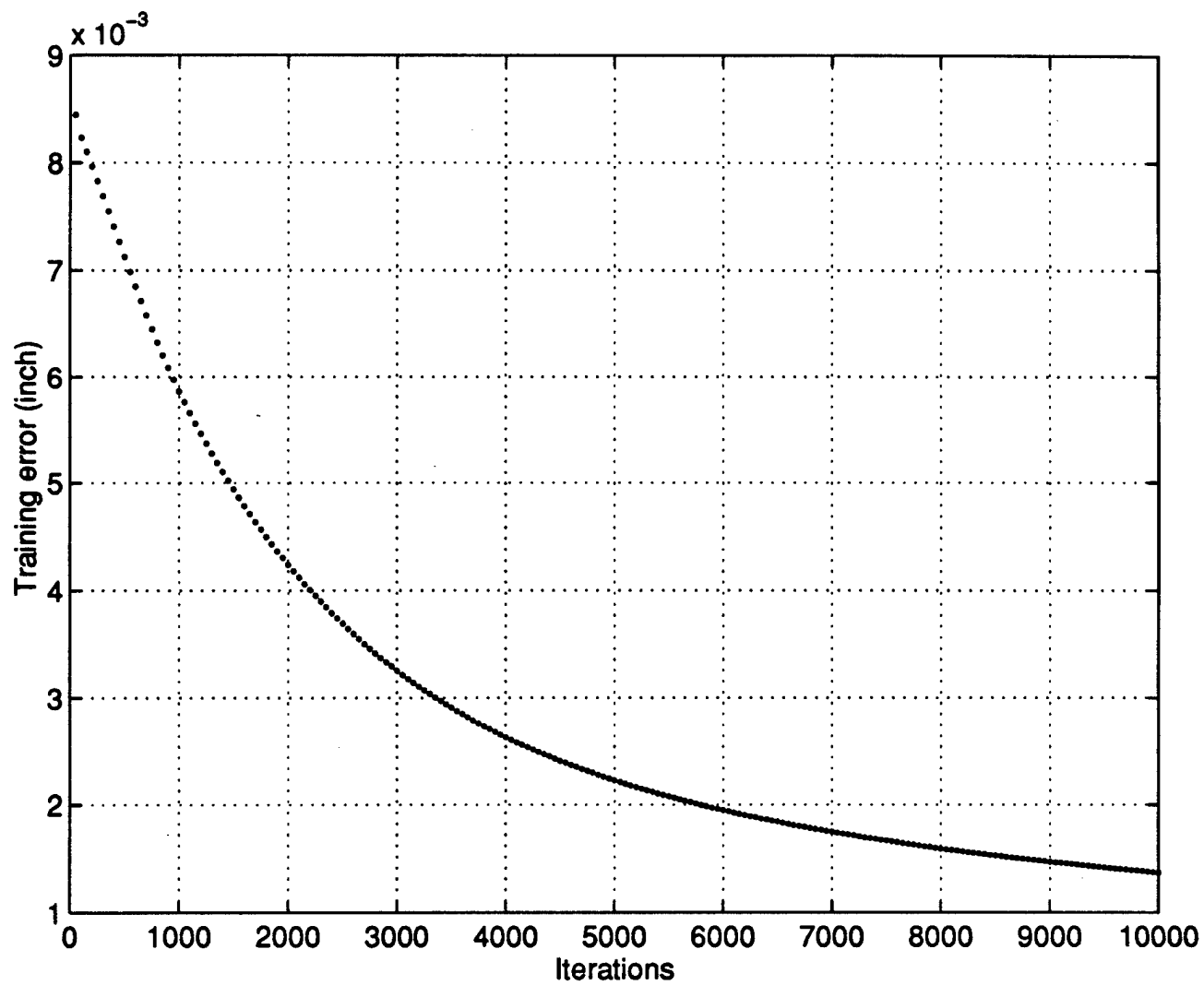
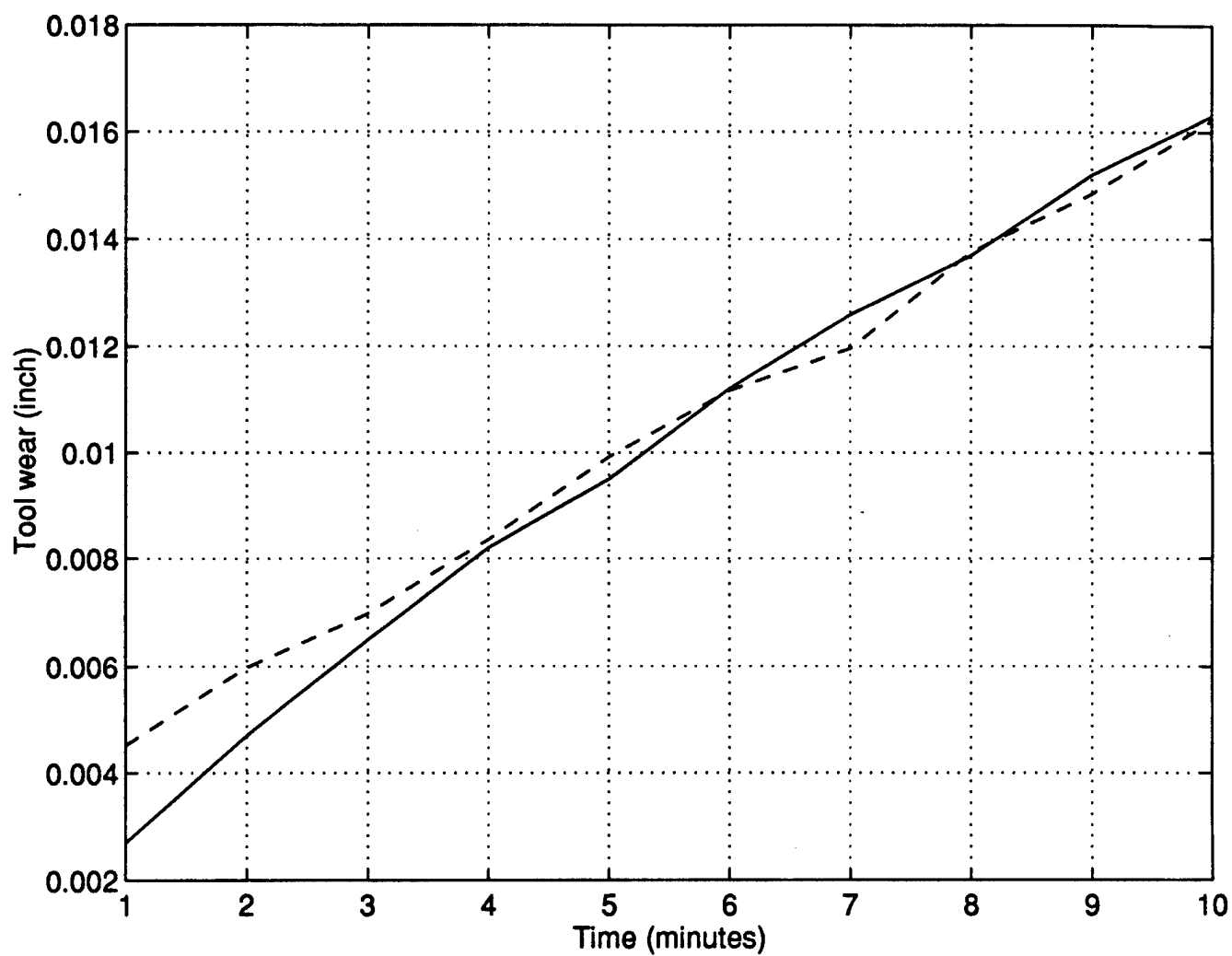Figure 13: Convergence of training errors.

Figure 14: Comparison of tool wear estimates from the fractal estimator (– – –) and the actual measured tool wear (——).
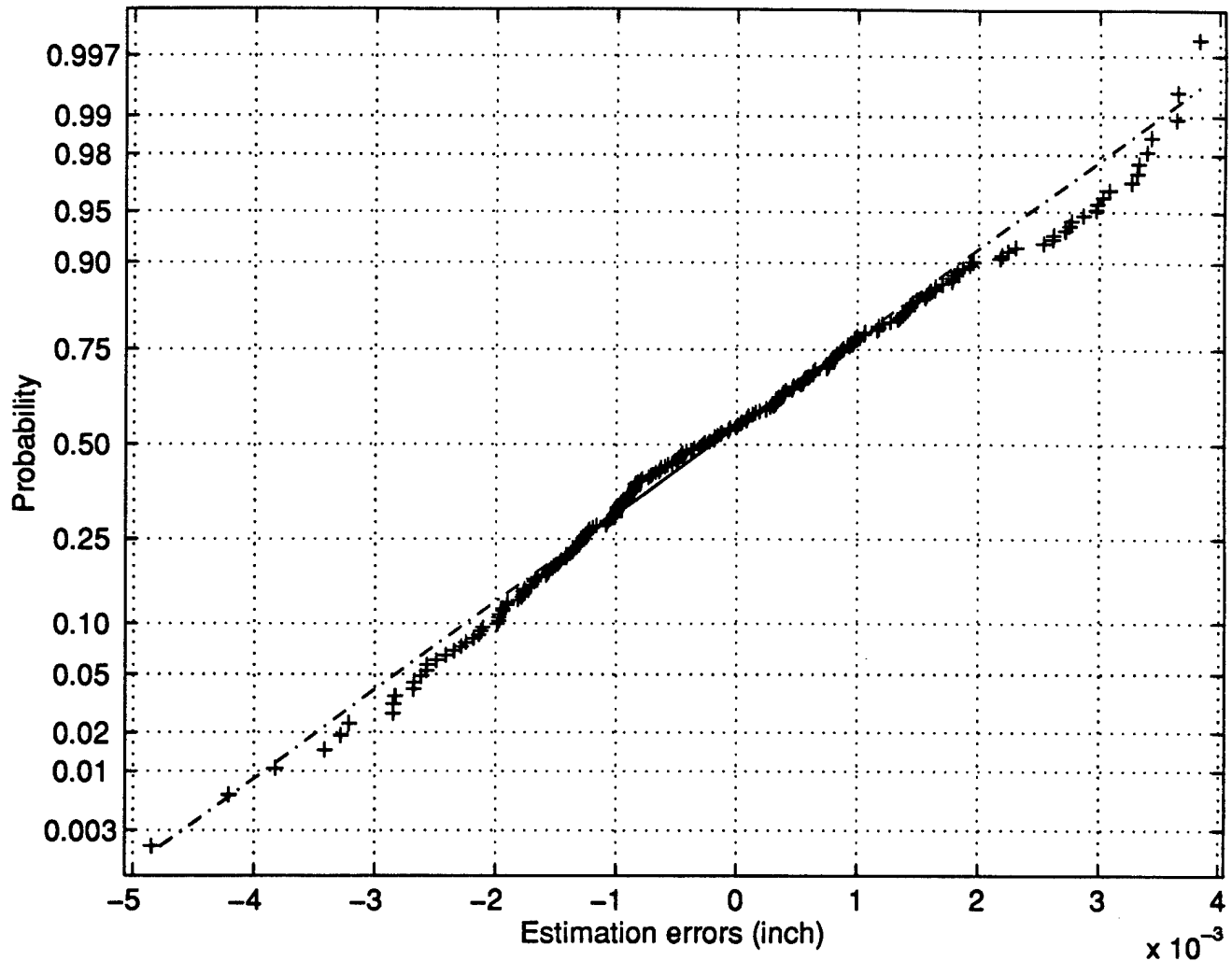
Figure 15: Normal probability plot of estimation errors.

of the estimation errors, computed using the t-statistic, was (−0.000076 inch, 0.000333 inch). The accuracy is comparable to that obtained from the radial basis network architecture (Kamarthi, 1994). Furthermore, the results suggest that fractal estimator possibly overestimates the flank wear values. But this overestimation has been found from Student t-test to be statistically insignificant at 95% confidence interval.

Results of some important performance evaluation tests are presented in the following subsections. In these tests, we report the effects of (a) the two process parameters, namely cutting speed and feed, and (b) actual tool wear on the estimates. Since the number of samples for tests are not significantly large, we used a Monte-Carlo simulations-based statistical inference method called *bootstrapping* (Efron, 1987). This bootstrap resampling technique improves the accuracy of the results of testing. Following these, we discuss the generalizability of fractal estimator.

3.1   Influence of cutting speed and feed on the estimates

Variation of estimation errors with respect to cutting speed and feed are depicted by the bubble plot, shown in Figure 16. The diameter of the bubble is proportional to the mean of estimation errors. A
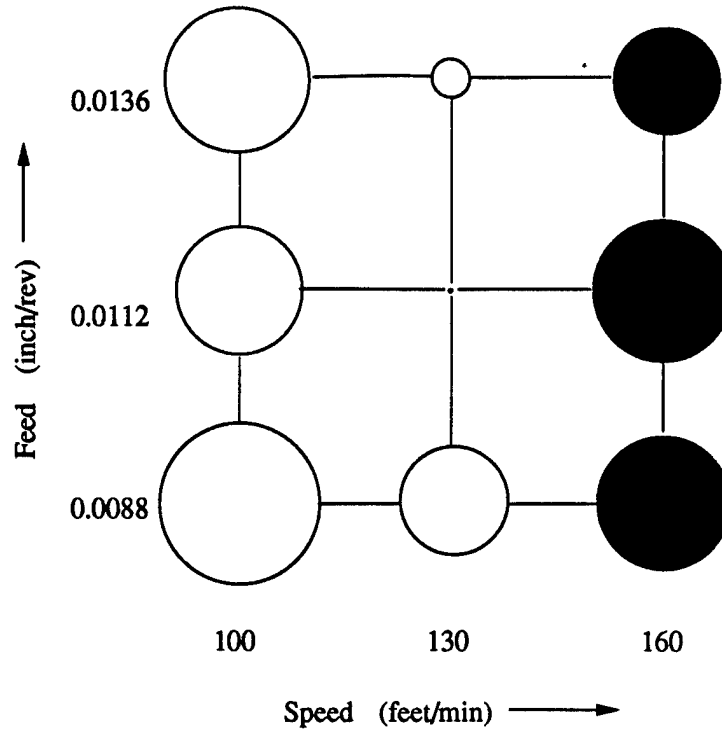
Figure 16: Influence of cutting speed and feed on the estimation error.

shaded bubble represents a negative value, which alludes to the underestimation of tool wear by the fractal estimator.

The bubble plot shows that the fractal estimator predominantly overestimates tool wear. But, at higher cutting speeds the fractal estimator tends to underestimate tool wear. This variation in the estimation—though statistically insignificant—can be of practical concern.

From the process standpoint, we have occasionally observed builtup edge at cutting speeds of 100 feet/min, and crater wear at 160 feet/min during our experimentation. Builtup edge tends to increase the magnitude of forces while crater wear has a tendency to decrease the force magnitudes. This is a plausible explanation for overestimation at low cutting speeds and underestimation at high cutting speeds.

3.2 Influence of tool wear on the estimates

The effect of tool wear on the estimation errors are shown in Figure 17. The figure shows that during the start of cutting, i.e., at low values of tool wear the fractal estimation overestimates flank wear. This implies that the current feature extraction scheme may be improved to capture the *run-in* stage of flank wear. During steady flank wear growth process, the estimates are reasonably accurate, implying the sufficiency of fractal estimator during the mild wear stage.

3.3 Generalizability of the fractal estimator

Generalizability refers to the ability of a fractal estimator to extrapolate the relations learnt on the training set to the patterns (testing patterns in the present case) presented during the operation phase. In order to understand the generalizability of fractal estimator, we compared the estimation errors obtained for the testing patterns against those for the exemplar patterns used for training. The results of our comparisons shown in Figure 18 revealed that the two distributions are reasonably close. This implies in a way that the fractal estimator for flank wear estimation has a fair degree of generalizability.
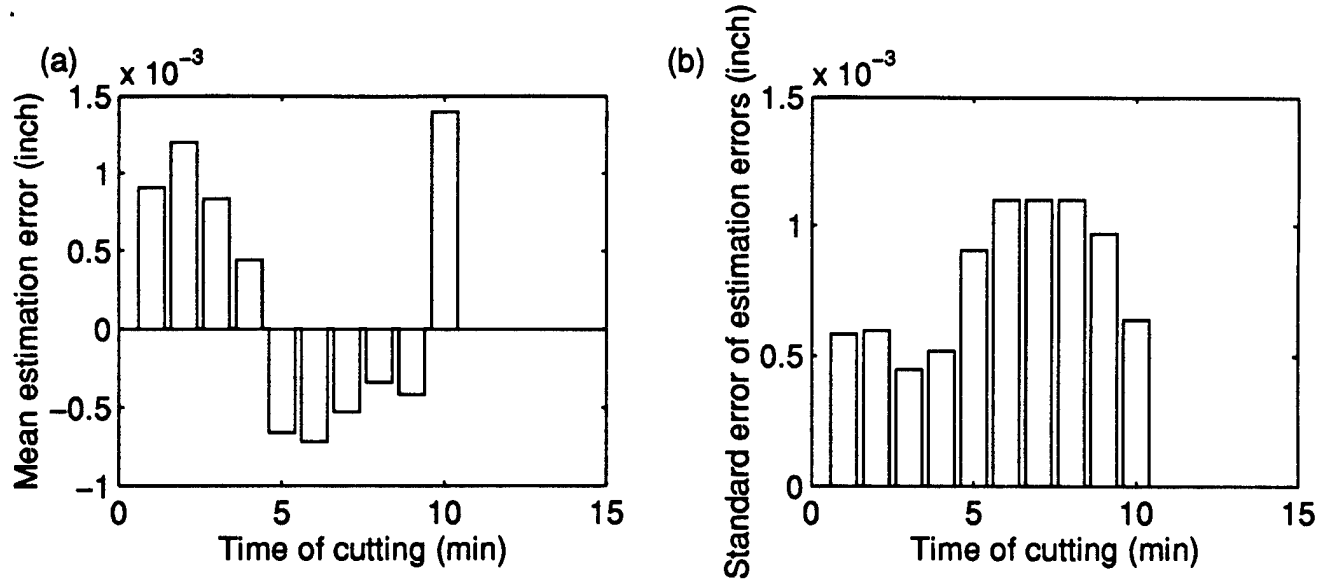
Figure 17: Influence of time of cutting, and hence the extent of flank wear, on the estimation error.

3.4 General remarks on the performance

Based on the results of the performance testing, we shall now summarize the performance capabilities of the fractal estimator in the following:

1. **Identifiability:** The accuracy of the fractal over the operating parameter range matches the desired requirement of < 0.0018 inch ($< 10\%$ of the total range). Based on this result the experimental design and the exemplar patterns extracted therefrom seem adequate. Furthermore, during our training phase, we conducted training using different initial weights sampled from a zero-mean unit variance normal distribution (standard normal distribution). We observed that the accuracy levels remained the same. These observations strongly suggest the identifiability of the fractal estimator.

2. **On-line implementation:** Since the stages of signal separation and feature extraction can be implemented in real-time, we may conclude that the fractal estimator, like any other neural network architecture, is implementable on-line. The on-line implementation issues are discussed in the next section.

3. **Stability:** In this work the stability of the estimator has not been rigorously explored. But stability of the state estimator was presumed and experimentally shown to be satisfactory.

4. **Dimensionality:** One main advantage of our scheme over the existing tool wear estimation schemes (Kamarthi, 1994) is the reduction in the dimension of the exemplar patterns. Due to lower dimensionality, our scheme enables a facile mapping of the neural network activations with the internal states of turning dynamics.

5. **Robustness:** The architecture of fractal estimator is fairly simple implying the existence of mathematically tractable relationships connecting fractal properties of the sensor signals and the tool wear estimates. Furthermore, since we extract the features from the outputs of turning dynamics (sensor signals), sampled at high sampling rates, the effect of environment variables such as temperature become marginalized. Therefore robustness-enhancing methods such as hybrid architectures may not be necessary. We however foresee the use of robustness-enhancing architectures in order to mitigate the effects of sensor failures, and the sensitivity of the estimates to tool wear, process parameters and extraneous phenomena.

For the first time in metal cutting literature, we select the features based on the fractal properties of the signal, and the state estimation is carried out therefrom. From the results of performance evaluation, it is evident that fractal estimation of tool wear is a viable methodology.
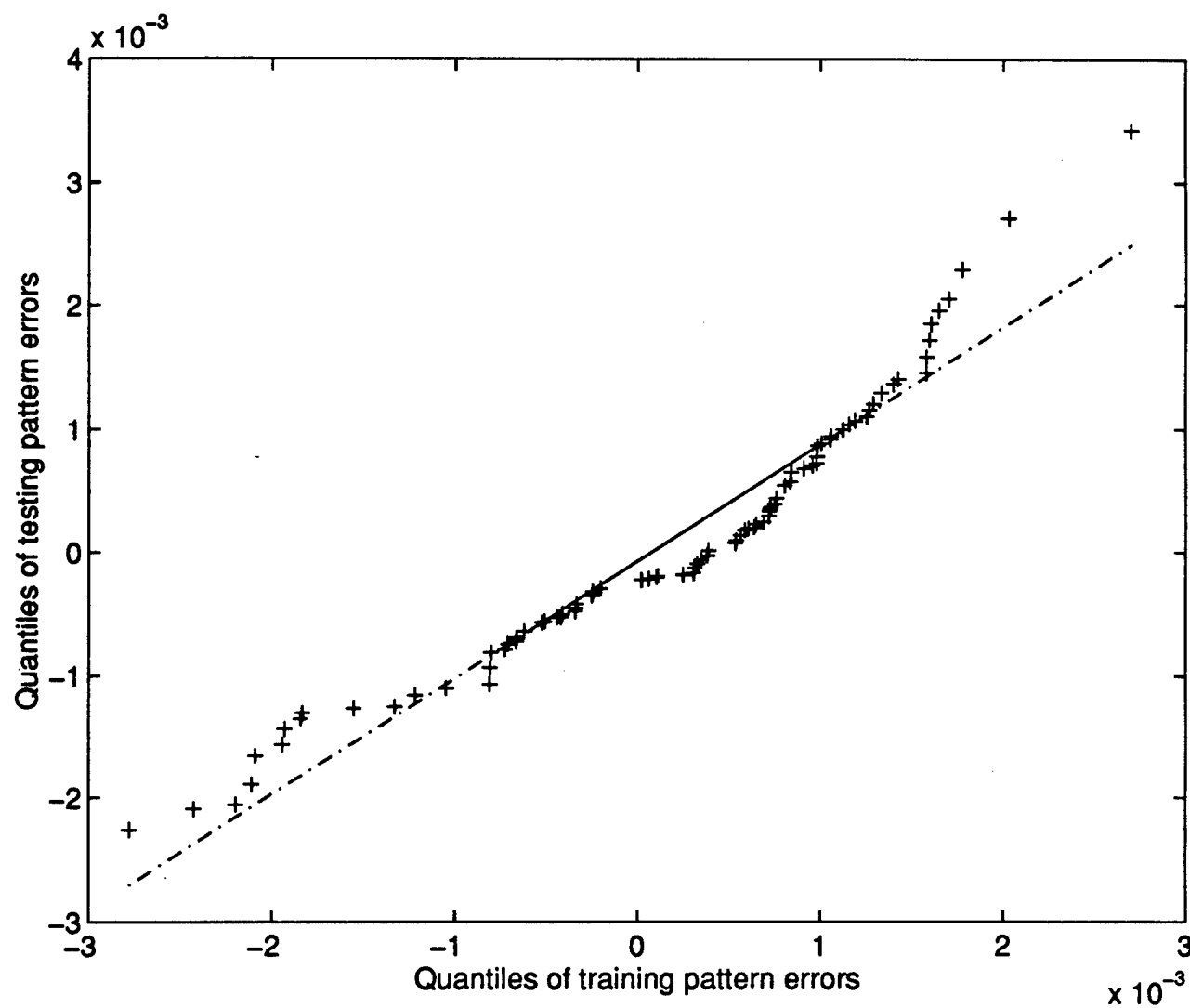
22

Figure 18: An empirical q-q plot showing the equivalence of the distributions of the estimation errors corresponding to the exemplar patterns used for training the neural network, and those corresponding to the testing patterns.

4 OPERATION PHASE ISSUES

During the operation phase, the flank wear estimates will be computed at regular time intervals t_1 and the measured TSD will be sampled at $t_2 \geq t_1$ intervals. We list below some of the operational issues we foresee in such a scenario:

Data acquisition: On-line sensor data shall be collected in separate channels, and digitized individually, similar to the experimentation described in this paper. Since the stability of the fractal estimator has not been thoroughly verified, we propose to use tool breakage, chatter and built-up edge, and sensor failure detectors as the *gates*. This will prevent a strong perturbation in the turning dynamics from affecting the flank wear estimates.

Signal separation: A moving time window $t_w = 4096 \times t_2$ may be used to compute the wavelet transform and the threshold. The window is updated at regular intervals of $t_3 \in (t_2, t_1)$, $t_3 \approx t_2$. The threshold value corresponding to the window in the previous instant is used for soft-thresholding the wavelet coefficients in the current time-instant. This schema will facilitate on-line signal separation.

Feature extraction: If there are atleast 4 good sensors, compute the fractal dimensions of the 4-dimensional vector TSD constructed therefrom, otherwise, use lag coordinates (Abarbanel et al., 1993) to obtain the 4-dimensional vector TSD. The computed fractal dimension values, in conjunction with the current process parameter values shall be inputted to the fractal estimator to obtain tool wear estimates at regular intervals of t_1.

The operation phase issues presented here are by no means exhaustive. More insight into the operation issues can be derived only through costly on-line implementation.

5 CONCLUSIONS

This two part paper has introduced a new methodology of tool wear estimation by associating the fractal properties of signal signals with continuous (gradual) flank wear through a recurrent neural network. We anticipate that this paradigm, based on combining fractal analysis and neural networks, will provide a new direction for not only the future research in the area of tool wear estimation, but also in estimation of gradual failure mechanisms.

In addition, we note that performance of fractal estimator has room for further improvement. We are currently developing a multifractal representation of tool wear, and investigating the possibility of an alternate neural network architecture to improve the fractal estimator performance.

Acknowledgments

References

Abarbanel, H., Brown, R., and Tsimiring, L., 1993, "The analysis of observed chaotic data in physical systems," *Reviews of Modern Physics*, Vol. 65, pp. 1331–1422.

Bukkapatnam, S. T. S., Lakhtakia, A., and Kumara, S. R. T., 1995, "Analysis of sensor signals shows that turning process on a lathe exhibits low-dimensional chaos," *Physical Review E*, Vol. 52, pp. 2375–2387.

Goldberg, A., 1993, "Applications of wavelets to quantization and random process representations," Ph. D. Thesis, Department of Electrical Engineering, Stanford University, Stanford, CA.

ISO, 1972, "Tool life testing with single point tools," ISO 5th Draft Proposal ISO/TC 29/WGG22 (Secretariate 37), Vol. 91.

Kamarthi, S. V., 1994, "On-line flank wear estimation in turning using multi-sensor fusion and neural networks," PhD Thesis, Department of Industrial and Manufacturing Engineering, University Park, PA.

Liebovitch, L., and Toth, T., 1989, "A fast algorithm to determine dimensions by box counting," *Physics Letters A*, Vol. 141, pp. 386–390.

Mallat, S. G., 1989, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions of Pattern Analysis and Machine Intelligence*, Vol. 11, pp. 674–693.

Zbikowski, R. W., 1994, "Recurrent neural networks: Some control aspects," PhD Thesis, Department of mechanical Engineering, Glasgow University, Glasgow, UK.